REMARKS

I. Introduction

In response to the Office Action dated April 8, 2005, claims 1, 5, 24, 39 and 43 have been amended. Claims 1-57 remain in the application. Re-examination and re-consideration of the application, as amended, is requested.

II. Claim Amendments

Applicants' attorney has made amendments to the claims as indicated above. These amendments were made solely for the purpose of clarifying the language of the claims, and were not required for patentability or to distinguish the claims over the prior art.

III. Non-Art Rejections

In paragraphs (3)-(4) of the Office Action, claims 5, 24, and 43 were rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Applicants' attorney has amended claims 5, 24 and 43 to overcome this rejection.

In paragraph (5) of the Office Action, claim 39 was rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Applicants' attorney has amended claim 39 to overcome this rejection.

In paragraphs (6)-(8) of the Office Action, claims 1-19 were rejected under 35 U.S.C. §101, because they are directed to non-statutory subject matter.

Applicants' attorney has amended claim 1 to overcome this rejection.

IV. Prior Art Rejections

A. The Office Action Rejections

In paragraph (10) of the Office Action, claims 1, 5, 6, 20, 24, 25, 39, 43, and 44 were rejected under 35 U.S.C. §103(a) as being unpatentable over IBM Technical Disclosure Bulletin "Resource Management System for Multimedia Devices" (IBM) in view of Arakawa et al, U.S. Publication No. 2002/0065793 (Arakawa). In paragraph (15) of the Office Action, claims 2-3, 21-22, and 40-41 were rejected under 35 U.S.C. §103(a) as being unpatentable over IBM in view of Arakawa, and further in view of Hintz et al, U.S. Patent No. 5,222,235 (Hintz). In paragraph (19) of the Office Action,

claims 4, 23, and 42 were rejected under 35 U.S.C. §103(a) as being unpatentable over IBM in view of Arakawa, and further in view of Bordonaro et al., U.S. Patent No. 5,307,485 (Bordonaro). In paragraph (22) of the Office Action, claims 7-11, 26-30, and 45-49 were rejected under 35 U.S.C. §103(a) as being unpatentable over IBM in view of Arakawa, and further in view of "Official Notice" (ON). However, in paragraph (24) of the Office Action, claims 12-19, 31-38 and 50-57 were indicated as being allowable if rewritten in independent form to include the base claim and any intervening claims.

Applicants' attorney acknowledges the indication of allowable claims, but respectfully traverse these rejections.

B. Applicants' Independent Claims

Applicants' independent claims 1, 20 and 39 are directed to loading data into a data store connected to a computer. Independent claim 1 is representative and comprises a computer-implemented method performing the steps of:

identifying memory constraints;

identifying processing capabilities; and

determining a number of load and sort processes to be started in parallel based on the identified memory constraints and processing capabilities.

C. The IBM Reference

IBM describes a resource management system for multimedia devices.

D. The Arakawa Reference

Arakawa describes a sorting system that includes a plurality of input nodes, each of which sorts sorting target data distributed and stored in input local disks. An internally sorted result is stored as a plurality of sorted strings in a shared disk connected between the input node and output node. Upon reception of a merge instruction from all input nodes, the output node reads the sorted string from the shared disk and merges it and outputs a whole sorted result of all input data to an output local disk. In a process of obtaining a whole sorted result of all input data through parallel processing by a computer system constituted of a plurality of computers (nodes), a time to sorting input data can be shortened.

E. The Hintz Reference

Hintz describes how the reorganization method of DB2 data files explores parallel processing, and asynchronous I/O to a great extent. It includes means to estimate an optimum configuration of system resources, such as storage devices (DASD devices), memory, and CPUs, etc, during reorganizations. The method mainly consists of four components, (1) concurrent indexing, (2) concurrent unloading of data file partitions, (3) efficient reloading of DB2 data pages and DB2 space maps, and (4) means to reduce access constraints to the DB2 recovery table.

F. The Bordonaro Reference

Bordonaro describes a system and method for merging a plurality of sorted lists using multiple processors having access to a common memory in which N sorted lists which may exceed the capacity of the common memory are merged in a parallel environment. Sorted lists from a storage device are loaded into common memory and are divided into a number of tasks equal to the number of available processors. The records assigned to each task are separately sorted, and used to form a single sorted list. A multi-processing environment takes advantage of its organization during the creation of the tasks, as well as during the actual sorting of the tasks.

G. Applicants' Independent Claims Are Patentable Over The References

Applicants' attorney respectfully submits that Applicants' claimed invention is patentable over the references. Specifically, Applicants' attorney asserts that the references do not teach or suggest the limitations recited in Applicants' independent claims 1, 20 and 39.

Nonetheless, the Office Action states the following:

> 11. Regarding claims 1, 20, and 39, IBM Bulletin discloses a method of processing data comprising the steps of:
> identifying memory constraints (page 4, 1st paragraph: number of concurrently executing tasks is bounded by memory);
> identifying processing capabilities (page 4, 1st paragraph: number of concurrently executing tasks is bounded by processor capabilities); and
> determining a number of tasks to be started in parallel based on the identified memory constraints and processing capabilities (page 4, 1st paragraph: number of concurrently executing tasks is bounded by memory and processor capabilities).
> IBM Bulletin did not disclose the tasks are the load and sort tasks. Nevertheless, Arakawa discloses of parallel processing, in which tasks such as sort processes and load processes performed in parallel (fig. 11, page 1, paragraph 12: executing parallel sorting processes at a plurality of computers. Page 6, paragraph 88.

Page 7, paragraph 92: a process of storing a sorted string from plurality of input nodes to plurality of shared disk 500 is executed). Arakawa further discloses that parallel processing shortens a process time by preparing a plurality of resources necessary for information processing and performing a plurality of tasks at the same time, thereby attempting to reduce the processing time (page 1, paragraph 8). It would have obvious for one of an ordinary skill in the art, at the time the invention was made, to incorporate the load and sort functions/tasks of Arakawa together with IBM Bulletin to concurrently perform tasks such as loading the audio and/or midi tasks and sorting the tasks according to the size of the data/file to take advantage of high speed processing (Arakawa: page 1, paragraph 8) for performance enhancement and optimization.

Applicants' attorney disagrees. The cited portions of these references do not teach or suggest the combined limitations of Applicants' claims.

For example, the cited portions are set forth below:

IBM Bulletin: Page 4, 1st paragraph

- For devices implemented with a (DSP), the number of concurrently executing DSP tasks is bounded by the DSP memory and processor capabilities. Resource unit allocation for devices implemented on a DSP might be the following:

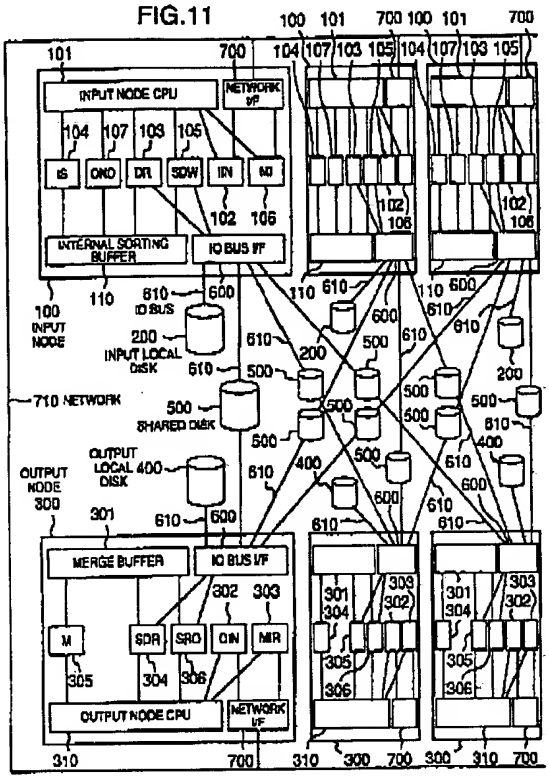
- FAX task - 10 resource units
- Audio task - 2 resource units
- MIDI task - 44 units

The problem of managing multiple logical devices implemented in hardware, such as a DSP or video transfer channel where hardware resources bound the number of device contexts of each device that can be concurrently supported, is addressed by varying the number of resource units each logical device can support. The multimedia resource manager provides an interface for the hardware device manager to specify the number of resource units a logical device can support.

When the number of resource units a logical device can support changes, the multimedia resource manager suspends or restores the device contexts for that logical device.

Arakawa: Fig. 11

FIG.11



Arakawa: Page 1, paragraph 12

[0012] It is an object of the present invention to provide a sorting system and method capable of executing parallel sorting processes at a plurality of computers in high speed by shortening the sorting process time.

Arakawa: Page 6, paragraph 88

[0088] After the management information necessary for the sorting process is prepared by the management information negotiation process, a sorting target data reading process is executed at each input node, in the manner similar to that of the first sorting system.

Arakawa: Page 7, paragraph 92

[0092] After the internal sorting process is completed in the above manner, a process of storing a sorted string in the shared disk 500 is executed. FIG. 14 is a flow chart illustrating this process of storing a sorted string in the shared disk 500. The shared disk writing unit 105 of each input node 100 selects the shared disk 500 connected to the output node 300 to which the internal sorted result stored in the internal sorting buffer 11 is output as the sorted string (S2040). Similar to that described for the first sorting system, the shared disk writing unit 105 generates a sorted string storing header for each selected shared disk 500 (S2050), and writes the

header and the internally sorted string into each selected shared disk 500 (S2060) to then update the sorted string storing information in each selected shared disk 500 (S2070).

Arakawa: Page 1, paragraph 8

[0008] Parallel processing is one type of high speed information processing to be executed by a computer system. Parallel processing shortens a process time by preparing a plurality of resources necessary for information processing and performing a plurality of tasks at the same time, thereby attempting to reduce the processing time. As an example of a parallelized external sorting process, JP-A-8-272545 discloses techniques of improving parallel processing by using a disk array constituted of a plurality of magnetic disks as an external storage and devising the storage locations of data in the disk array.

The above portions of the prior art references, when combined, do not teach or suggest all the claim limitations of Applicants' claims.

IBM merely describes the number of Fax, Audio and MIDI tasks that can be started on a single digital signal processor (DSP) based on the DSP memory and processor capabilities.

Arakawa merely describes executing parallel sorting processes at a plurality of computers in high speed in order to shorten the sorting process time.

However, the IBM and Arakawa references, taken individually or in combination, do not teach or suggest the limitations of "identifying memory constraints," "identifying processing capabilities," and "determining a number of load and sort processes to be started in parallel based on the identified memory constraints and processing capabilities." Indeed, this combination of functions is not taught or suggested by either reference. Consequently, it cannot be said that the combination of IBM and Arakawa teaches or suggests, or renders obvious, the Applicants' independent claims.

In addition, the Office Action fails to establish a prima facie case of obviousness, because there is no suggestion or motivation, in either reference themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the references or to combine the teachings of the references. See, e.g., M.P.E.P. §§ 706.02(j) and 2142. Instead, the suggestion comes from the Office Action itself, which constitutes impermissible hindsight.

In this regard, there must be a reasonable expectation of success, which is not present. IBM and Arakawa are directed at two fields of endeavor, i.e., resource management for multimedia devices (IBM) and parallel sorting of data (Arakawa). Because of these differences, there can be no expectation of success from the combination.

Finally, Hintz, Bordonaro and ON fail to overcome the deficiencies of IBM and Arakawa. Recall that Hintz was cited only against dependent claims 2-3, 21-22 and 40-41, Bordonaro was cited only against dependent claims 4, 23 and 42, and ON was cited only against dependent claims 7-11, 26-30 and 45-49. Moreover, Hintz was cited only for teaching determining the number of build processes based on the number of sort processes and the number of sort processes does not exceed the number of indexes built. In addition, Bordonaro was cited only for teaching that the number of load processes does not exceed a number of partitions to be loaded, and that the load and sort processes directly dependent on memory constraints. Finally, ON was cited only for teaching to efficiently utilize all processing capabilities required for the desired task. None of these teachings are relevant to the limitations of Applicants' independent claims.

Thus, Applicants' attorney submits that independent claims 1, 20 and 39 are allowable over the references.

H. Applicants' Dependent Claims Are Patentable Over The References

Dependent claims 2-11, 21-30 and 40-49 are submitted to be allowable over the references in the same manner, because they are dependent on independent claims 1, 20 and 39, respectively, and thus contain all the limitations of independent claims 1, 20 and 39. In addition, dependent claims 2-6, 21-25 and 40-45 recite additional novel elements not shown by the references.

With regard to claims 2, 21 and 40, which recite "determining a number of build processes based on the number of sort processes," the Office Action asserts that these limitations are described in Hintz at Fig. 1 and col. 3, line 46 – col. 4, line 7. Applicants' attorney disagrees. At the indicated location, Hintz merely describes multitasking and a database containing multiple table spaces or indexes, but says nothing about determining the number of build processes based on the number of sort processes.

With regard to claims 3, 22 and 41, which recite that "the number of sort processes does not exceed a number of indexes to be built," the Office Action asserts that these limitations are described in Hintz at col. 5, lines 50-51. Applicants' attorney disagrees. At the indicated location, Hintz merely describes a method for building indexes from a sorted index work file, one index at a time, but says nothing about the number of sort processes not exceeding the number of indexes to be built.

With regard to claims 4, 23 and 42, which recite that "the number of load processes does not exceed a number of partitions to be loaded," the Office Action asserts that these limitations are described in Bordonaro in FIG. 3 (310 and 312), FIG. 2 (202), at col. 4, line 62 – col. 6, line 27, and col.

5, lines 58-60. Applicants' attorney disagrees. At the indicated locations, Bordonaro merely describes merging M lists residing in common memory to create a final sorted list.

With regard to claims 5, 24 and 43, which recite that "the total number of load and sort processes does not exceed processing capabilities," the Office Action asserts that these limitations are described in IBM at page 4, 1st paragraph. Applicants' attorney disagrees. At the indicated location, IBM merely states that the number of concurrently executing DSP tasks is bounded by the DSP memory and processor capabilities.

With regard to claims 6, 25 and 44, which recite that "the memory utilized by the load and sort processes does not exceed memory constraints," the Office Action asserts that these limitations are described in IBM at page 4, 1st paragraph. Applicants' attorney disagrees. At the indicated location, IBM merely states that the number of concurrently executing DSP tasks is bounded by the DSP memory and processor capabilities.

With regard to claims 7, 26 and 45, which recite that "the number of load processes and the number of sort processes each require different processing power," these claims stand or fall with independent claims 1, 20 and 39, respectively.

With regard to claims 8, 27 and 46, which recite that "the number of load processes and the number of sort processes each require similar processing power," these claims stand or fall with independent claims 1, 20 and 39, respectively.

With regard to claims 9, 28 and 47, which recite that "the number of load processes is not equal to the number of sort processes," these claims stand or fall with independent claims 1, 20 and 39, respectively.

With regard to claims 10, 29 and 48, which recite that "the number of load processes is equal to the number of sort processes," these claims stand or fall with independent claims 1, 20 and 39, respectively.

With regard to claims 11, 30 and 49, which recite that "the number of load processes is equal to the number of sort processes and which is equal to half of the processing capabilities," these claims stand or fall with independent claims 1, 20 and 39, respectively.

V. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited.

Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicants' undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicants

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: July 5, 2005

By: _____
Name: George H. Gates
Reg. No.: 33,500

GHG/bjs

G&C 30571.279-US-01